REMARKS/ARGUMENTS

Pending claims 1, 8, 13, 17 and 22 stand rejected under 35 U.S.C. § 101 as allegedly being directed to non-statutory subject matter. Applicant respectfully disagrees. Claim 1 has been amended to recite the execution of one or more instructions of the second thread in a processor after a switch, and the storage of the results. Accordingly, a physical transformation and a concrete and substantial final result exists and the rejection of claim 1 is overcome. For similar reasons amended claim 8 defines patentable subject matter, as it recites execution of at least one instruction of the second thread after a switch to reduce or prevent a stall.

As to claim 13, it is directed to a computer-readable medium that stores instructions that are executed to enable a system to perform particular operations. As such, the recited subject matter of the claim is clearly within one of the four categories of subject matter to which a claim can be directed (process, machine, manufacture, or composition of matter). Accordingly, under the Interim Guidelines, claim 13 and its dependent claims recite statutory subject matter. That is, the recited computer-readable medium stores instructions and as such, the claims are directed to a statutory category of subject matter.

As to the §101 rejection of claims 17 and 22, Applicant first notes that the claims recite "a processor pipeline." Such a processor pipeline is clearly hardware. Furthermore, both claims recite a feedback loop, i.e., hardware. Still further, claims 17 and 22 recite that a first pipeline stage (in claim 17) and an instruction decoder unit (in claim 22) generate a feedback signal that is provided along this feedback loop. Such tangible hardware that generates such a signal is clearly statutory subject matter. This clear hardware recitation can in no way be construed to be "a timing relationship" such as a "trigger in the timing diagram," as contended by the Office Action. Instead, both claims 17 and 22 set forth detailed structural elements of the recited apparatus and system. Accordingly, the rejection of these claims under §101 is overcome.

Pending claims 1-3, 8, 10, 11 and 17 stand rejected under 35 U.S.C. § 102(b) over U.S. Patent No. 6,049,867 (Eickemeyer). Applicant respectfully traverses the rejection. As to claim 1, Eickemeyer nowhere teaches determining whether instruction execution potentially causes a long latency. Instead, Eickemeyer teaches that if a level two cache miss or translation lookaside buffer miss occurs, threads are switched. These events are not a potential cause of a long latency, but rather an actual long latency. Nor does Eickemeyer anywhere teach or suggest that a determination be made as to whether an instruction potentially causes a long latency. Instead, all

that Eickemeyer teaches is that if either of these two events occur, a thread switch may occur. Thus there is no determination in Eickemeyer as to whether an instruction potentially causes a long latency, and in fact there is no analysis of latencies whatsoever in Eickemeyer. For at least these reasons, claim 1 and its dependent claims are patentable over Eickemeyer.

Dependent claim 2 is further patentable as Eickemeyer nowhere teaches execution of at least one additional instruction in the first thread while preparing to switch to a second thread. In this regard, the Office Action simply refers to saving a thread's state or determining enabling of thread switching. Office Action, page 6. However, Eickemeyer does not teach that such actions are instructions of the first thread. Instead, these are performed outside of the thread. E.g., Eickemeyer, Figure 3 (note that execution of thread i in block 86 is separate and independent of determining if thread switching is enabled or saving of thread i state in blocks 90 and 92).

Dependent claim 3 is further patentable as Eickemeyer nowhere teaches a determination based on a stochastic modeling. As defined in Applicant's specification, a stochastic modeling is a statistical basis, i.e., a knowledge that a condition "may, but need not necessarily," cause a latency. Specification, page 8, lns. 12-16. The valid bit of Eickemeyer is not a stochastic model. Instead, the valid bit of Eickemeyer is just that, a bit to indicate whether a given thread is in a valid state. Eickemeyer, 4:58-61. Such a bit is not a modeling whatsoever, and certainly is not a stochastic modeling. Accordingly, for this further reason claim 3 is patentable. For at least the respective same reasons, claims 8, 10 and 11 are also patentable over Eickemeyer.

Claim 17 is patentable over Eickemeyer, as Eickemeyer nowhere teaches a feedback loop coupled between first and second pipeline stages to provide a feedback signal generated in the second stage back to the first stage. In this regard, the Office Action merely refers to a flowchart of Eickemeyer that discloses passing control from a first thread to a second thread. However, nowhere does this method teach a feedback signal that is generated in a second stage, nor a feedback loop coupled between multiple pipeline stages to provide the feedback signal between the stages. Accordingly, claim 17 is patentable over Eickemeyer.

Pending claims 1-6, 8-9, 11-14, 16-21, and 30 stand rejected under §102(b) over U.S. Patent 6,076,157 (Borkenhagen). Applicant respectfully traverses the rejection. As to claim 1, Borkenhagen similarly nowhere teaches determining whether instruction execution potentially causes a long latency. Instead, Borkenhagen causes thread switches based on actual inability to execute an instruction of a thread. Borkenhagen, column 16, lines 38-51. This actual failure to

execute is not a potential cause of a long latency. Accordingly, claim 1 and the claims depending therefrom are patentable over Borkenhagen.

As to dependent claim 4, Borkenhagen further nowhere teaches that an instruction is applied to a lookup table that includes entries corresponding to predetermined conditions. Instead, the cited portions of Borkenhagen (Office Action, page 4) nowhere teach or suggest applying an instruction to a lookup table. Instead, all that this portion of Borkenhagen teaches is that a translation lookaside buffer includes virtual-to-real address mappings. However, such mappings are not entries corresponding to predetermined conditions, nor does Borkenhagen anywhere teach or suggest applying an instruction to a lookup table to determine whether execution of the given instruction potentially causes a long latency. Certainly, Borkenhagen nowhere further teaches or suggests providing a feedback signal if this instruction matches an entry in the lookup table, as recited by claim 5. As to dependent claim 30, Borkenhagen nowhere teaches that a lookup table includes entries corresponding to predetermined conditions where the predetermined conditions comprise instruction types. Instead, all that the cited portion of Borkenhagen teaches (Borkenhagen, 3:21-33) is that different types of processors execute instructions that may be completed in-order and out-of-order. This nowhere teaches either predetermined conditions, nor a lookup table including entries corresponding to such predetermined conditions. For this further reason, dependent claim 30 is patentable.

For similar reasons described above regarding claim 1, the rejection of claims 8 and 13 over Borkenhagen is overcome, as Borkenhagen nowhere teaches thread switching if a condition that could potentially result in a stall occurs. Also, for the same reason discussed above regarding claim 3, the rejection of dependent claim 11 is improper as Borkenhagen nowhere teaches basing a condition on a stochastic model.

Dependent claim 12 is further patentable, as Borkenhagen nowhere teaches providing a feedback signal generated in an instruction decoder to an instruction fetch unit. That is, for similar reasons discussed above regarding claim 5, Borkenhagen further nowhere teaches comparison of an instruction to entries in a lookup table to determine whether a condition that potentially results in a processor pipeline stall occurs.

As to claim 17, the citation of a flowchart in Borkenhagen nowhere teaches the recited feedback loop coupled between multiple processor pipelines stages, as clearly this software

executed flowchart does not teach or suggest such a hardware feedback loop between multiple pipeline stages. Accordingly, claim 17 and its dependent claims are patentable.

Claims 22 and 24-29 stand rejected under 35 U.S.C. § 103(a) over Borkenhagen in view of U.S. Patent Number 6,076,157 (Rompaey). This rejection is improper, at least for the failure of Borkenhagen to teach or suggest the claimed feedback loop coupled to provide a feedback signal generated an instruction decoder to an instruction fetch unit. In this regard, the Office Action contends that "Borkenhagen also taught a feedback from an instruction decoder to fetch unit." (Office Action, p. 4, referring to a feedback signal from a translation lookaside buffer to a sequencer). However, Borkenhagen merely teaches that the translation lookaside buffer stores virtual-to-real address mappings and is in no way an instruction decoder. Furthermore, the cited portion of Borkenhagen in column 9 merely teaches that the sequencers output signals to a thread switch logic. However, this thread switch logic is neither an instruction decoder nor an instruction fetch unit. Instead, Borkenhagen teaches that the thread switch logic "contains various registers that determine which thread will be the active or the executing thread." (Borkenhagen, 8:17-19) – not instruction fetching or decoding. Nor does Rompaey remedy this failure.

The rejection is further improper as there is no basis to combine Borkenhagen with Rompaey. In this regard, Rompaey is directed to non-analogous art. MPEP § 2141.01. Instead of the multi-threaded processor of Borkenhagen, Rompaey is directed to co-design of hardware and software. The subject matter of Rompaey simply has no bearing on the claimed subject matter, or that of Borkenhagen. Rompaey describes in its background that:

> Digital communication techniques form the basis of the rapid breakthrough of modern consumer electronics, wireless and wired voice-and data networking products, broadband networks and multi-media applications. Such products are based on digital communications systems, which are made possible by the combination of VLSI technology and digital signal processing.

Rompaey, col. 1, lns. 18-24.

This is the alleged motivation to combine Rompaey with the thread switching of Borkenhagen. Clearly, this portion (along with the rest of Rompaey) nowhere provides a motivation to combine its teaching with that of Borkenhagen. MPEP § 2143.01. Similarly, Borkenhagen fails to provide a motivation to combine its teaching with the hardware/software co-design of Rompaey. Accordingly, the proposed combination is improper and the rejection is overcome.

New dependent claim 31 is patentable at least for at least the same reason as claim 5, discussed above.

For at least the same reasons described above with regard to the independent claims, the various dependent claims rejected under § 103(a) over varying combinations are also patentable.

In view of these remarks, the application is now in condition for allowance and the Examiner's prompt action in accordance therewith is respectfully requested. The Commissioner is authorized to charge any additional fees or credit any overpayment to Deposit Account No. 20-1504.

Respectfully submitted,

Date: October 20, 2006

Mark J. Rozman
Registration No. 42,117
TROP, PRUNER & HU, P.C.
1616 S. Voss Road, Suite 750
Houston, Texas 77057-2631
(512) 418-9944 [Phone]
(713) 468-8883 [Fax]
Customer No.: 21906